TRANSMISSION CONTROL FOR MINIMIZING CONGESTION IN DIGITAL COMMUNICATIONS NETWORKS

Technical Field

The invention relates to transmissions in a digital communications network and, more specifically, to transmission control for minimizing network congestion.

Background of the Invention

For preventing loss of data due to congestion in digital network communications, a protocol known as Transmission Control Protocol (TCP) has been proposed for the Internet; see Information Sciences Institute, "Transmission Control Protocol - Request for Comments 793", September 1981 and W. Stevens, "TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms - Request for Comments 2001", January 1997. TCP is based on the notion of fair sharing of transmission resources among users.

TCP is reliable, in the sense that the data received at a destination are an exact duplicate of the data that was sent. Such reliability may be at the expense of transmission delays, however.

For some transmissions, e.g. real-time audio and video, reliability is less important, and the primary concern is with the data arriving on time. Specifically, for example, it is more acceptable to lose an occasional frame of video than to have the video start and stop repeatedly.

Summary of the Invention

For congestion control in a digital communications network such as the Internet or corporate "Intranets", and especially for real-time transmissions in such networks,

Fig. 6 is a representation of packet format for a preferred embodiment of the invention in a wireless or hybrid wired-wireless network.

Detailed Description of Preferred Embodiments

While preferred embodiments are described in the following primarily in method terms, the inventive technique also includes systems embodiments, e.g. involving a programmed processor. A prototype implementation uses a Unix Workstation as network server and a PC as client server, both programmed in C++. Use of special-purpose firmware or hardware is not precluded.

The technique is window-based in the sense that a sender maintains a count of the number of outstanding packets, i.e., packets which have been sent, but for which an acknowledgment has not yet been received from the receiver. The sender maintains current an upper bound on the number of outstanding packets allowed in the network, called the "congestion window" (CWND) and providing an indication of the available bandwidth from sender to receiver. Congestion is detected when a packet is lost in the network. Alternatively, and especially in transmissions of variable-length packets, CWND can be maintained in units of bytes rather than units of packets.

If the number of outstanding packets is less than CWND, the sender can continue to send data into the network. Otherwise, the sender must stop transmitting data until either CWND increases or the number of outstanding packets decreases. If acknowledgments are received, CWND will increase, and the number of outstanding packets will decrease. If no acknowledgments are returned, packets will timeout and be deemed lost by

"Outstanding acknowledgments" (ACK) is set to zero. "Timeout" (TO) is set to 3 seconds, for example, indicating the amount of time not to be exceeded between sending a packet and receiving its acknowledgment. If an acknowledgment is not received in time, the packet is assumed to be lost. The system starts out in a "Slow-Start Phase" indicated by Phase=SS.

Since CWND is the size of the first packet, ACK=0, and there is data available to send (namely the first packet), the first packet is sent into the network. ACK is then increased by the size of the packet sent, representing the number of bytes currently in the network that have not yet been acknowledged. The system then checks whether acknowledgments have arrived. If so, Outstanding Acknowledgments is decreased by the size of the packet to which the acknowledgment refers: ACK = ACK-size. The system then calculates the Round Trip Time (RTT), i.e. the difference between when a packet was sent and when the acknowledgment was received. RTT is used in the calculation of Timeout (TO).

The system maintains an estimate of the round trip time, RTT_{avg}, by using the measured RTT, RTT_i, for each acknowledgment. Following D. Comer, "Internetworking with TCP/IP", 3rd Edition, Simon & Schuster, 1995, pp. 191-230, RTT_{avg} and Timeout (for future use) are calculated as follows:

$$Diff = RTT_i - RTT_{avg}$$
$$RTT_{avg} = RTT_{avg} + Diff/8$$
$$Dev_i = 0.25 \cdot (|Diff| - Dev_i)$$
$$Timeout = RTT + 0.25 + 3 \cdot Dev_i$$

Now, in Slow Start Phase, CWND is increased by *size*:

$$CWND = CWND + size;$$

fills a buffer. At the server, the media pump sends the data to the client from the buffer, taking into account the current value of CWND determined in accordance with Fig. 2, and the media pump supplies the size values for congestion control. In case of significant congestion, CWND will be less than ACK, and this will stop the media pump from sending further data for a period of time, thereby reducing the media pump transmission rate.

So long as the average available bandwidth of a connection is greater than or equal to the bandwidth requirements of the media, and so long as there is sufficient buffering, the media can be played back without interruption. With congestion-minimizing processing as described above, few packets will be lost, and can be retransmitted if there is enough time.

Buffering provides for variation in the available bandwidth: the larger the buffer, the more variation can be accommodated. But there is an initial start-up delay while a client buffer is being filled, so that increased buffering results in a longer start-up delay.

As to adaptable media, there are several ways of changing bandwidth requirements. In the case of MPEG, for example, one way involves dropping frames as described by Z. Chen et al., "Real Time Video and Audio in the World Wide Web", World Wide Web Journal, Vol. 1, January 1996. The server finds the picture header in the MPEG stream and stops sending data until it finds the next picture header in the stream. This has the effect of dropping one frame from the media stream, and thereby reducing the bandwidth requirements. As frames are interdependent in MPEG, a frame should not be dropped if other frames depend on it, i.e. an I-frame cannot be dropped if the stream contains P- or B-frames which depend on it.

media being adapted. The media pump operates as in the non-adaptable case, sending data only when CWND > ACK. Based on the occupancy of the buffer, the adaptable media module is instructed to change the rate of the media.

For example, for rate control in MPEG video by frame dropping, a frame can be dropped when the buffer is more than half full; otherwise, the video is passed unaltered to the buffer. Other scenarios, using DRS and more sophisticated rate control may be implemented. For example, if the buffer is filling, the transmission rate may be reduced in inverse relationship to the rate of buffer filling.

Fig. 4 illustrates an exemplary rate control technique based on measurements of buffer occupancy. Every 5 seconds, an average buffer occupancy is obtained for the previous 5 seconds, $Occupancy_i$. The change in the buffer occupancy since the previous 5-second interval, $Occupancy_{i-1}$, is determined as $Diff_i$, Start-up is with $Occupancy_0 = 0$.

The Centering factor provides a weighting for the occupancy to stay close to the desired occupancy at the buffer midpoint. The maximum buffer size is 5 seconds worth of data and depends on the originally encoded rate of the stream.

If $Diff_i < 0$,

$$Centering_i = Occupancy_i/Occupancy_{desired},$$

where $Occupancy_{desired}$ is the buffer occupancy which rate control tries to maintain. Otherwise,

$$Centering_i = 2 - (Occupancy_i/Occupancy_{desired}),$$

the goal being to keep the Centering factor between 0 and 2.

Then, $Beta_i$ is determined as a direct indication of how much demand varies in the network, using the Coefficient of Variation of the past and current values

the bandwidth requirements of the media can be reduced down to a minimum of 150 kbps. When the available bandwidth drops to 200 kbps, the media also is reduced to this rate, so that no receiver buffering is used to compensate for the network. However, once the available bandwidth decreases to 100 kbps, the media can only be reduced to 150 kbps, and so the receiver buffer begins to be depleted. This scenario is more robust, as the available bandwidth can drop to 150 kbps and receiver buffering is not used.

Congestion control in accordance with the invention is applicable wherever some degree of loss can be tolerated, including most video and audio codecs, with adaptable codecs being preferred. Most video codecs can be adapted by using frame dropping. Even still images can be adapted for real-time applications. JPEG and MPEG have similarities in the way they are coded, so that a technique like DRS can be used on JPEG as well. A new standard known as Flashpix has the capability to be displayed at different resolutions, and hence different bandwidth requirements when sending a picture across the Internet.

While preferred embodiments have been described above under the assumption of a wired network, composed of fiber-optic or coaxial physical cables, techniques of the invention can be used to advantage with wireless networks as well. As digital communications protocols were originally devised with wired networks in mind, most congestion-aware protocols, TCP included, assume that a lost packet indicates congestion. This is practicable in wired networks, where bit errors are uncommon. Bit errors are more common in a wireless environment, however, so that a packet is more likely to become "lost" due to an error in the packet, regardless of congestion. But known systems do not include facilities for informing

application drops the packet and sends a request for retransmission to the sender— without invoking congestion avoidance to reduce the transmission rate at the sender. If there is no error, the packet is used by the receiver application, with regular acknowledgment.

In this fashion, the likelihood of a packet being dropped by the receiver operating system due to packet error is minimized, and greater throughput is realized on wireless networks without impairing the performance on wired networks. No changes are required to the operating system nor the underlying network link layer, so long as the link layer does not perform error checking over the entire link layer packet.

This preferred technique can be used with all proprietary client-server protocols which are congestion-aware. Such protocols must be proprietary because of changes to both the client and the server. Accordingly, adaptable media applications are preferred.